REMARKS

In the Office Action, the Examiner rejected claims 1-21 and 23-24 under 35 U.S.C. §103(a) as being unpatentable by U.S. Patent No. 6,212,640 issued to Abdelnur et al. ("Abdelnur") in view of U.S. Patent No. 6,615,276 to Mastrianni ("Mastrianni"). In this Amendment, Applicant has amended claims 1, 2, 8, 19, and 24. Applicant has amended these claims for reasons of clarity and not for reasons of patentability. Applicant does not surrender any equivalents of the amended elements. Applicant has canceled claim 9. Applicant has not added any new claims. Accordingly, claims 1-8, 10-21, and 23-24 will be pending after entry of this Amendment.

I. Claims 1-8 and 10-18

In the Office Action, the Examiner rejected claims 1-8 and 10-18 under §103(a) as being unpatentable by Abdelnur in view of Mastrianni. Claims 2-8 and 10-18 are directly or indirectly dependent on claim 1. Claim 1 recites a method for automatically mounting several remote volumes to a client. The method, without receiving a request from a user of the client to mount the remote volumes, logs into a first server. The method, via a plug-in module installed on the client, receives a set of mounting parameters from the first server. The method, at the client, automatically mounts the volumes by utilizing the returned parameters. The volumes after mounting behave as native to the client, even though the volumes are located remote to the client.

Applicant respectfully submits that the cited references, neither alone nor through their piecemeal, hindsight combination, make claim 1 unpatentable for at least the following reasons. The cited references, neither alone nor combined, disclose, teach, or even suggest a method, that at the client, automatically mounts the remote volumes by utilizing a set of parameters received from a server.

In the Office Action, the Examiner has cited Abdelnur as specifying such a limitation. *See*, page 2 of the Office Action, which cites column 5, lines 61-67 and column 6, lines 1-15 of Abdelnur for automatically mounting remote volumes by utilizing parameters returned from the first server. Applicant respectfully submits that the cited references do not disclose, teach, or even suggest a method that (1) receives a set of mounting parameters from a server and (2) automatically mounts a set of volumes by utilizing the received parameters. Rather, Abdelnur discloses that a remote server mounts a volume, generates a set of results, and returns the generated results to the client.

Specifically, the cited paragraphs of Abdelnur disclose a Network File System (NFS). Abdelnur discloses:

> An NFS server is a computer that shares its resources with other computers (NFS clients) on the network, using the NFS service. *See,* Abdelnur, column 6, lines 4- 6.
>
> Using NFS, a resource (i.e., software) physically linked to a NFS server may be "NFS mounted." The resource that is "NFS mounted" is accessible to all NFS clients as if the software were stored locally at the client. *See,* Abdelnur, column 6, lines 7-10.

Applicant respectfully submits that in Abdelnur the NFS mounts the volumes. The NFS system is used by an application to make a request for a procedure to be performed and the results to be returned. Specifically, Abdelnur discloses:

> Using NFS, application makes for a procedure to be performed, (e.g., a read or write request). *See,* Abdelnur column 6, lines 33-37.
>
> While application is waiting for a response to its request, NFS client encodes the contents of the local request into a remote-procedure-call (RPC). *See,* Abdelnur column 6, lines 40-42.
>
> Once NFS server receives a RPC request from NFS client, the request is decoded and processed as a local file system operation. The result generated by NFS server is encoded and returned to NFS client. (Emphasis added). *See,* Abdelnur, column 6, lines 53-56.

The NFS method disclosed in Abdelnur generates the results and returns it to the client as opposed to the method of claim 1 in which the server returns a set of mounting parameters. The returned parameters are utilized to automatically mount the volumes. Applicant respectfully submits that returning a result by a server is different than returning a set of mounting parameters and mounting of the volume by the client. Therefore, Applicant respectfully submit that the cited references do not disclose, teach, or even suggest (1) receiving from a first server a set of mounting parameters, and (2) automatically mounting a set of volumes by utilizing the received parameters as recited in claim 1.

Accordingly, Applicant respectfully submits that Abdelnur and Mastrianni, neither alone nor through their piecemeal, hindsight combination, render claim 1 invalid. Given that claims 2-8 and 10-18 are dependent on claim 1, Applicant respectfully submits that claims 2-8 and 10-18 are allowable over the cited reference for at least the same reasons that were provided above for claim 1. In view of the foregoing, Applicants respectfully request reconsideration and withdrawal of the 103(a) rejection of claims 1-8 and 10-18.

II. Claims 19-21

In the Office Action, the Examiner rejected claims 19-21 under §103(a) as being unpatentable by Abdelnur in view of Mastrianni. Claims 20 and 21 are directly or indirectly dependent on claim 19. Claim 19 recites a system for automatically mounting volumes over a network. The system includes a client that is configured to automatically request the mounting of the volumes without the need for a user of the client to request the mounting of the volumes. The system also includes a server that is coupled to the client. The server is configured to return a mounting command block to the client. The client utilizes the returned mounting command block to mount the volumes.

Applicant respectfully submits that the cited references, neither alone nor through their piecemeal, hindsight combination, make claim 19 unpatentable for at least the following reasons. The cited references, neither alone nor combined, disclose, teach, or even suggest a system for automatically mounting volumes over a network in which a server returns a mounting command block to a client.

In the Office Action, the Examiner has cited Abdelnur as specifying such a limitation. *See*, page 7 of the Office Action, which cites column 5, lines 61-67 and column 6, lines 1-15 of Abdelnur for a system with a server configured to return a mounting command block to a client. Applicant respectfully submits that the cited references do not disclose, teach, or even suggest a system for automatically mounting volumes over a network in which a server returns a mounting command block to a client. Rather, Abdelnur discloses that a remote server mounts a volume, generates a set of results, and returns the generated results to the client.

Specifically, the cited paragraphs of Abdelnur disclose a Network File System (NFS). Abdelnur discloses:

> An NFS server is a computer that shares its resources with other computers (NFS clients) on the network, using the NFS service. *See,* Abdelnur, column 6, lines 4- 6.
>
> Using NFS, a resource (i.e., software) physically linked to a NFS server may be "NFS mounted." The resource that is "NFS mounted" is accessible to all NFS clients as if the software were stored locally at the client. *See,* Abdelnur, column 6, lines 7-10.

Applicant respectfully submits that in Abdelnur the NFS mounts the volumes. The NFS system is used by an application to make a request for <u>a procedure to be performed and the results to be returned.</u> Specifically, Abdelnur discloses:

> Using NFS, application makes for a procedure to be performed, (e.g., a read or write request). *See,* Abdelnur column 6, lines 33-37.
>
> While application is waiting for a response to its request, NFS client encodes the contents of the local request into a remote-procedure-call (RPC). *See,* Abdelnur column 6, lines 40-42.
>
> Once NFS server receives a RPC request from NFS client, the request is decoded and processed as a local file system operation. The <u>result generated</u> by NFS server <u>is encoded and returned to NFS client</u>. (Emphasis added). *See,* Abdelnur, column 6, lines 53-56.

The NFS method disclosed in Abdelnur generates the results and returns it to the client as opposed to the method of claim 19 in which the server returns a mounting command block to the client. Applicant respectfully submits that returning a result by a server is different than returning a mounting command block to the client.

Accordingly, Applicant respectfully submits that Abdelnur and Mastrianni, neither alone nor through their piecemeal, hindsight combination, render claim 19 invalid. Given that claims 20 and 21 are dependent on claim 19, Applicant respectfully submits that claims 20 and 21 are allowable over the cited reference for at least the same reasons that were provided above for claim 19. In view of the foregoing, Applicants respectfully request reconsideration and withdrawal of the 103(a) rejection of claims 19-21.

III. Claim 23

In the Office Action, the Examiner rejected claim 23 under §103(a) as being unpatentable by Abdelnur in view of Mastrianni. . Claim 23 recites an apparatus in a computer system that has a processor and memory. The apparatus can automatically mount several remote volumes to the computer system. The apparatus includes a plug-in module that is configured to receive a mounting command block from a server. The mounting command block includes volume mounting parameters. The apparatus also includes an application programming interface coupled to the plug-in module and configured to automatically mount the volumes specified in the mounting command block. The mounting is performed without the need for user intervention.

Applicant respectfully submits that the cited references, neither alone nor through their piecemeal, hindsight combination, make claim 23 unpatentable for at least the following reasons. *First*, the cited references, neither alone nor combined, disclose, teach, or even suggest a plug-in module that is configured to receive a mounting command block that includes parameters from a server. In the Office Action, the Examiner has cited column 5, lines 5-40 of Abdelnur for disclosing "a plug-in module configured to receive from a server a mounting command block, said mounting command block to include volume mounting parameters. *See*, page 8 of the Office Action.

Applicant respectfully submits that the cited paragraphs of Abdelnur disclose a Java applet and not a plug-in module:

> The classes of a Java applet are loaded on demand from the network (stored on a server), or from a local file system, when first referenced during the Java applet's execution. *See,* Abdelnur column 5, lines 17-19.

For at least the following reasons, Applicant respectfully submits that Abdelnur's Java applet is not the recited plug-in module of claim 23. As is known in the art, plug-in applications are programs that are installed and used as part of a client application (such as a web browser) and are not limited to use the resources of the server from which they are loaded. Also, the Java applet disclosed in Abdelnur is not for receiving a mounting command block from the server when the mounting command block includes mounting parameters. Instead, it is for accessing the resources on the server. For instance, column 5, lines 29-34 of Abdelnur states:

> A Java applet loaded from the network server is executed on the client's virtual machine. An applet has limited permission to access the resources available on the server and other network computers. In prior art schemes, this access is typically limited to the resources available on the server where the applet is loaded from.

In view of the foregoing, Applicant respectfully submits that the cited references do not disclose, teach, or suggest the apparatus of claim 23 that includes a plug-in module that is configured to receive a mounting command block that includes mounting parameters from a server.

Second, Applicant respectfully submit that the cited references neither alone nor through their piecemeal, hindsight combination disclose, teach, or event suggest an application programming interface (API) coupled to a plug-in module and configured to automatically mount a set of remote volumes specified in a mounting command block. The application programming interface disclosed in Abdelnur is utilized by applications to interface with an NFS client and is not an API connected to a plug-in.

Third, as discussed in Sections I above, the NFS method disclosed in Abdelnur, generates the results and returns it to the client as opposed to the computer system of claim 23 in which (1) the server returns a mounting command block and (2) the application programming interface (that is connected to the plug-in module) automatically mounts the volumes specified in the mounting command block. Applicant respectfully submits that the cited references do not disclose, teach, or even suggest an apparatus that includes an application programming interface coupled to a plug-in module and configured to automatically mount a set of remote volumes specified in a mounting command block.

Accordingly, Applicant respectfully submits that Abdelnur and Mastrianni, neither alone nor through their piecemeal, hindsight combination, render claim 23 invalid. In view of the foregoing, Applicants respectfully request reconsideration and withdrawal of the 103(a) rejection of claim 23.

IV. Claim 24

In the Office Action, the Examiner rejected claim 24 under §103(a) as being unpatentable by Abdelnur in view of Mastrianni. . Claim 24 recites an apparatus for automatically mounting several volumes to a client. The apparatus includes means for requesting from a server the mounting of the volumes without receiving a request from a user of the client to mount the volumes. The apparatus also includes means for returning from the server a set of mounting parameters. The apparatus also includes means for automatically mounting the volumes at the client by utilizing the returned parameters. The volumes after mounting behave as native to the client, even though the volumes are located remote to the client.

Applicant respectfully submits that the cited references, neither alone nor through their piecemeal, hindsight combination, make claim 24 unpatentable for at least the following reasons. The cited references, neither alone nor combined, disclose, teach, or even suggest an apparatus with means that, at the client, automatically mounts the remote volumes by utilizing a set of parameters returned from a server.

In the Office Action, the Examiner has cited Abdelnur as specifying such a limitation. *See*, page 2 of the Office Action, which cites column 5, lines 61-67 and column 6, lines 1-15 of Abdelnur for automatically mounting remote volumes by utilizing parameters returned from the server. Applicant respectfully submits that the cited references do not disclose, teach, or even suggest an apparatus with means that (1) return a set of mounting parameters from a server and (2) automatically mount a set of volumes at a client by utilizing the returned parameters. Rather, Abdelnur discloses that a remote server mounts a volume, generates a set of results, and returns the generated results to the client.

Specifically, the cited paragraphs of Abdelnur disclose a Network File System (NFS). Abdelnur discloses:

> An NFS server is a computer that shares its resources with other computers (NFS clients) on the network, using the NFS service. *See,* Abdelnur, column 6, lines 4- 6.
>
> Using NFS, a resource (i.e., software) physically linked to a NFS server may be "NFS mounted." The resource that is "NFS mounted" is accessible to all NFS clients as if the software were stored locally at the client. *See,* Abdelnur, column 6, lines 7-10.

Applicant respectfully submits that in Abdelnur the NFS mounts the volumes. The NFS system is used by an application to make a request for a procedure to be performed and the results to be returned. Specifically, Abdelnur discloses:

> Using NFS, application makes for a procedure to be performed, (e.g., a read or write request). *See,* Abdelnur column 6, lines 33-37.
>
> While application is waiting for a response to its request, NFS client encodes the contents of the local request into a remote-procedure-call (RPC). *See,* Abdelnur column 6, lines 40-42.
>
> Once NFS server receives a RPC request from NFS client, the request is decoded and processed as a local file system operation. The result generated by NFS server is encoded and returned to NFS client. (Emphasis added). *See,* Abdelnur, column 6, lines 53-56.

The NFS method disclosed in Abdelnur generates the results and returns it to the client as opposed to the method of claim 1 in which the server returns a set of mounting parameters. The returned parameters are utilized to automatically mount the volumes. Applicant respectfully submits that returning a result by a server is different than returning a set of mounting parameters and mounting of the volume at the client. Therefore, Applicant respectfully submit that the cited references do not disclose, teach, or even suggest (1) returning from a server a set of mounting parameters, and (2) automatically mounting a set of volumes at a client by utilizing the returned parameters as recited in claim 1.

Accordingly, Applicant respectfully submits that Abdelnur and Mastrianni, neither alone nor through their piecemeal, hindsight combination, render claim 24 invalid. In view of the foregoing, Applicants respectfully request reconsideration and withdrawal of the 103(a) rejection of claim 24.

CONCLUSION

In view of the foregoing, it is submitted that all pending claims, namely claims 1-8, 10-21, and 23-24 are in condition for allowance. Reconsideration of the rejections and objections is requested. Allowance is earnestly solicited at the earliest possible date.

Respectfully submitted,

STATTLER, JOHANSEN & ADELI LLP

Dated: 8/7/06

Ali Makoui

Reg. No. 45,536

Stattler Johansen & Adeli LLP
1875 Century Park East, Suite 1360
Century City, CA 90067-2514
Phone: (310) 785-0140 x302
Fax: (310) 785-9558